We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



11

Mass Conservative Domain Decomposition for Porous Media Flow

Jan M. Nordbotten, Eirik Keilegavlen and Andreas Sandvin
University of Bergen
Norway

1. Introduction

Understanding flow in subsurface porous media is of great importance for society due to applications such as energy extraction and waste disposal. The governing equations for subsurface flow are a set of non-linear partial differential equations of mixed elliptic-hyperbolic type, and the parameter fields are highly heterogeneous with characteristic features on a continuum of length scales. This calls for robust discretization methods that balance the challenges in designing efficient and accurate methods. In this chapter we focus on a class of linear solvers for elliptic systems that aims at providing fast approximate solutions, preferably in one iteration, but fall back to being iterative methods with good convergence properties if higher accuracy is needed.

We consider flow of flow of a single fluid in a porous media, transporting a passive particle. This can for instance represent flow of a pollutant as a result of groundwater contamination. Governing equations for the flow will be presented in the next section. Analytical solutions to the flow problem can only be found in very special cases, and in general, numerical approximations must be sought. The primary numerical schemes for commercial simulations are control volume methods. These methods are formulated such that conservation of mass is ensured, which is considered crucial in applications. After discretizing, an elliptic equation needs be solved for the pressure. This process is computationally expensive and may constitute the majority of the simulation time.

The permeability (fluid conductivity of the rock) in subsurface porous media has a truly multiscale nature, with highly permeable pathways with significant correlation lengths. Hence the elliptic pressure equation will experience strong non-local effects, posing a challenge for linear solvers. Moreover, the permeability field constructed by geologists is highly detailed; the number of cells in the geo-model can easily be several orders of magnitude higher than what is feasible to handle in a flow simulation. The traditional approach to this problem has been to upscale the permeability, e.g. to compute a representative permeability on a coarser grid. For the pressure equation this gives a linear system that is much smaller and computationally cheaper to solve. The drawback is of course that details in the geological characterization may be lost during upscaling, and these details are known to have significant impact on transport. An alternative approach is offered by the so-called multiscale methods, which have been a highly popular research field in the last decade (Tchelepi & Juanes, 2008). Like upscaling, multiscale methods perform a coarsening to end up with a relatively small

linear system to solve. However, a multiscale method also provides a mapping from the coarse solution onto the fine grid. This projected solution will not be equal to a direct solution on the fine grid, but the two solutions will share many properties; in particular, many multiscale methods provide a velocity field that is mass conservative on the fine scale. Hence it can be used to solve fine scale transport equations. Numerical experiments have shown that this strategy can be extremely effective and highly accurate when measured in metrics that are important for applications (Kippe, et al., 2008; Efendiev & Hou, 2009).

Despite the success of multiscale methods in porous media flow, the strategy has certain weaknesses. In this chapter we highlight the quality of the coarse operator: If this does not represent essential features of the flow field, the quality of the fine scale velocity field may be poor. In particular, long and high permeable pathways are difficult to capture in the coarse scale operator. A natural approach would therefore be to introduce a scheme that allows for iterations on the multiscale solution. The idea of a multi-level iterative method resembles domain decomposition, and in Nordbotten & Bjørstad, 2008, the equivalence between the multiscale finite volume method (Jenny, et al., 2003) and a special domain decomposition strategy was shown. The resulting iterative scheme was termed mass conservative domain decomposition (MCDD), and it can be classified as an additive Schwarz preconditioner with minimal overlap. Contrary to classical domain decomposition methods, MCDD will produce solutions that are mass conservative at any iteration step, thus it is not necessary to reduce the pressure residual to a very low value before solving transport equations. Various aspects of MCDD have been tested for two-dimensional problems (Kippe, et al., 2008; Sandvin, et al., 2011; Lunati, et al., 2011). However, to formulate multiscale methods for three-dimensional problems has turned out to be considerably more difficult in general, and to our knowledge, no applications of MCDD-type methods within an iterative setting have been reported in three dimensions.

In this chapter, we consider multiscale methods and preconditioners defined for arbitrary number of spatial dimensions. We show how the multiscale method can be formulated both as a top-down and as a bottom-up method, and that these formulations give rise to different interpretations of the resulting approximations and preconditioners. Numerical examples illustrate the main strengths and weaknesses of the approach. Moreover, the numerical examples highlight the capabilities of the framework in terms of producing quick calculations when possible, but also producing more accurate results when needed.

2. Governing equations and discretization

The primary focus of the current chapter is linear solvers. The particular linear solvers we discuss are designed to preserve certain properties from the physical problem. Therefore, the linear solvers cannot be discussed without first specifying both the governing equations and the particular discretizations we are concerned with.

2.1 Governing equations

We consider flow of an incompressible fluid in a porous medium. For an introduction to flows in porous media, see e.g. (Bear 1972); for a reference focusing on appropriate numerical methods for this problem, confer (Chen, et al., 2006). Here we will only provide a brief review of the main ideas of importance to this chapter. Conservation of mass (volume) for each phase can be modeled by the equation:

$$\nabla \cdot \boldsymbol{u} = v.$$

Here the flux of each phase is represented by $\boldsymbol{u}$, and v denotes the volumetric source / sink terms. The flux is usually assumed to be given by Darcy's law, which reads

$$\boldsymbol{u} = -\boldsymbol{k}\nabla p,$$

where the permeability is denoted $\boldsymbol{k}$, and p is the fluid potential. Additionally, we consider a dissolved concentration c which is passively advected with the velocity field,

$$\phi \frac{\partial c}{\partial t} + \nabla \cdot (c\boldsymbol{u}) = \psi, \tag{1}$$

where ϕ is the fraction of the void space available for fluid flow, referred to as porosity, and the material source term is given by ψ. We note that by introducing the particle velocity $\boldsymbol{v} = \phi^{-1}\boldsymbol{u}$ the advection of the dissolved concentration can be written in terms of the material derivative on Lagrangian form

$$\frac{Dc}{Dt} = \frac{\psi}{\phi} - c\frac{\nabla \cdot \boldsymbol{u}}{\phi}.$$

By eliminating the fluid flux from the statement of volume conservation we obtain an elliptic equation for pressure

$$-\nabla \cdot (\boldsymbol{k}\nabla p) = v. \tag{2}$$

In the sequel, we will study numerical solution techniques for Eq. (2), while keeping in mind that the methods must also be applicable for complex situations. Specifically, by integrating the Lagrangian version of the transport equation, we see that volume balance errors lead to exponential growth of errors in the dissolved concentration. Thus, it is of importance for the problems we consider that the velocity field must always be mass conservative in order to be suitable for use with most transport schemes.

2.2 Discretization

To discretize Eq. (1), we consider scalar discretizations, and in particular control volume schemes as they are particularly well suited for an exact representation of the conservation equation. Introducing the usual L_2 inner product, we can write the elliptic equation on a weak form as: Find $p \in H_0^1$ such that

$$(\boldsymbol{k}\nabla p, \nabla w) = (v, w) \quad \text{for all } w \in W. \tag{3}$$

Here, W represents a suitably chosen space, and we have for simplicity assumed zero Dirichlet boundary conditions to simplify the exposition. From this equation, we obtain control volume methods by choosing the finite subset of $W_h \in W$ to be the piece-wise constants forming a partition of unity on a cell-based grid, from which we obtain

$$\int_{\partial\omega} \boldsymbol{u} \cdot \boldsymbol{n}\, d\sigma = \int_{\omega} v\, d\tau,$$

for each (primal) cell ω. Note that $\boldsymbol{n}$ is the unit normal vector pointing out from the cell, and the product $\boldsymbol{u} \cdot \boldsymbol{n}$ is the normal flux over the boundary. Various control volume methods can

now be defined by their approximation to the (flux) boundary integrals, most of which can be interpreted as particular choices of the finite space for p_h. We will in the following assume that such a choice has been made (for concreteness, one may consider the control-volume finite element method which is defined by p_h lying in the space spanned by piece-wise linears with nodes forming a dual grid to the partition induced by W_h). Furthermore, we assume for simplicity that the choice of flux approximation leads to a local approximation of the flux, in the sense that fluxes can be explicitly represented as a combination of fluid potentials in near-by cells.

We have now described a general setting for discrete representations of volume balance and Darcy's law which lead to a sparse linear system for the scalar variable p_h, which can be given on vector-matrix form as

$$\boldsymbol{Ap} = \boldsymbol{v}. \tag{4}$$

Remark 1: The control-volume finite element method, while attractive for educational purposes, is not very accurate in practice. Therefore, in reservoir simulation, the flux over a face has traditionally been approximated as driven by the pressure difference in the two adjacent cells only, giving rice to two-point schemes for the flux (Aziz & Settari, 1979). For logically Cartesian grids, this gives a classical 5- or 7-point cell stencil in 2 and 3 dimensions, respectively. However, in situations when the principal axis of the permeability tensor deviates considerably from the orientation of the grid, two-point schemes are known to produce inaccurate results. As a remedy, so-called multi-point schemes have been introduced (Edwards & Rodgers, 1998; Aavatsmark, 2002). These produce more accurate results to the price of a larger computational stencil, for Cartesian grids, the resulting linear system will have 9 and 27 bands in 2 and 3 dimensions, respectively. As we will see, the reduced accuracy of two-point schemes for rough grids is not only important for discretizing Eq. (2); similar considerations are also important when constructing fast linear solvers.

Remark 2: If a method is defined by choosing both p_h and w_h to lie in the same finite-dimensional space, the classical finite element method is recovered. In particular, the simplest choice, the piece-wise (multi)-linear functions give a system of equations that have a similar algebraic structure to the control-volume methods discussed above, but do not explicitly represent conservation.

3. Mass conservative domain decomposition

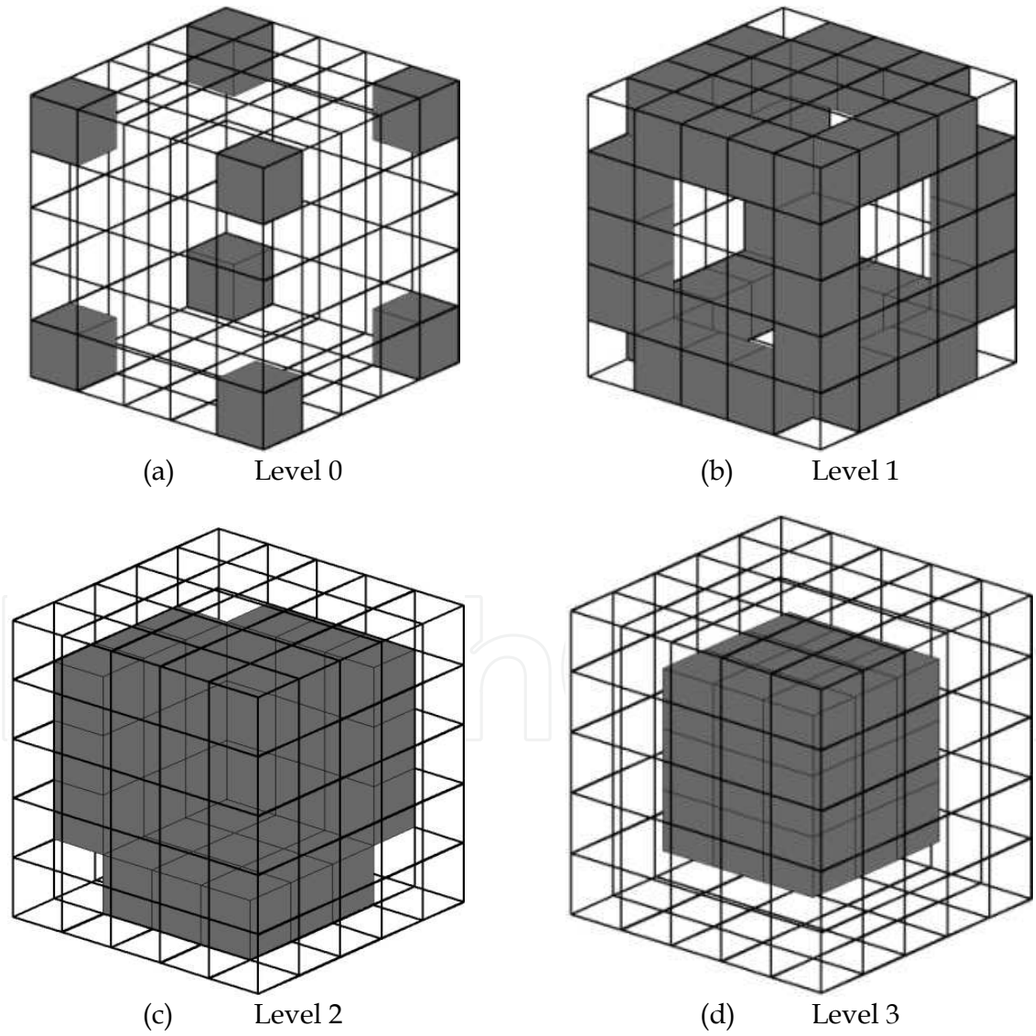
Here we describe the ingredients for the mass conservative domain decomposition (MCDD). We will start our presentation with describing the development of Schur-complement systems for N-dimensional problems. Readers may of course chose to disregard this generality, and consider only the special cases $N = 2, 3$. From this, we will see how we can form classical domain decomposition methods as well as multiscale control-volume methods. Note that due to the large number of matrices and vectors involved in the presentation, these will no longer be marked as bold as long as there is no room for confusion.

3.1 Schur complement systems

Being consistent with the discretization outlined above, we assume that computational domain is partitioned into a fine scale grid, and that control volume discretization enforces

conservation of mass on the fine grid cells. In order to proceed in the construction of a two-level method, we need to introduce the notion of coarse grids.

Consider a continuous collection of cells, referred to as internal boundary cells, which partition the domain into isolated subdomains. By isolated, we mean in the sense of the discretization of the elliptic operator, such that no cell in one subdomain is dependent on any cells of any other subdomains. We interpret the subdomains as cells of the coarse dual grid, and the internal boundary cells thus form the nodes, edges, faces etc. of the dual coarse grid. We will identify cells and variables with a numerical subscript dependent on what part of the dual coarse grid they form part of: 0 indicates dual coarse nodes, 1 indicates dual coarse edges, 2 indicates faces, and so on, until N denotes cells that lie in the subdomains (where N is the dimension of the problem). Confer Fig. 1 for an illustration. When considering only the internal boundary cells, we will refer with subscript B to all subscripts less than N. Note that we have not yet introduced a coarse primal grid; this will not be needed before a later section.



(a) Level 0 (b) Level 1

(c) Level 2 (d) Level 3

Fig. 1. Illustration of cells on different levels in a three-dimensional Cartesian grid. For clarity of visualization, only some of the cells on level 2 are indicated.

We now start to manipulate the linear system of equations (4), with the ultimate goal of obtaining a coarse linear system that captures non-local structures. By a reordering of the unknowns based on the dual coarse grid, Eq. (4) can then be written as

$$\begin{pmatrix} A_{00} & \cdots & A_{0N} \\ \vdots & \ddots & \vdots \\ A_{N0} & \cdots & A_{NN} \end{pmatrix} \begin{pmatrix} p_0 \\ \vdots \\ p_N \end{pmatrix} = \begin{pmatrix} v_0 \\ \vdots \\ v_N \end{pmatrix}. \tag{5}$$

In the last row, by construction, A_{NN} is a sparse block diagonal matrix, with each block representing the interactions within each isolated subdomain. This implies that we can find the values p_N by a local calculation given the variables on the internal boundary cells. We write these local calculations as

$$p_N = A_{NN}^{-1}(v_N - A_{NB}p_B).$$

We use this expression to formally eliminate internal cells from our system of equations. Thus, by substitution into Eq. (5) we have the *Schur Complement* system

$$\begin{pmatrix} S_{00} & \cdots & S_{0(N-1)} \\ \vdots & \ddots & \vdots \\ S_{(N-1)0} & \cdots & S_{(N-1)(N-1)} \end{pmatrix} \begin{pmatrix} p_0 \\ \vdots \\ p_{(N-1)} \end{pmatrix} = \begin{pmatrix} \tilde{v}_0 \\ \vdots \\ \tilde{v}_{(N-1)} \end{pmatrix},$$

where the Schur complement matrices S_{ij} are defined as

$$S_{ij} \equiv A_{ij} - A_{iN}A_{NN}^{-1}A_{Nj},$$

and the right hand side has been updated to reflect the elimination of the internal nodes by

$$\tilde{v}_i \equiv v_i - A_{iN}A_{NN}^{-1}v_N.$$

We make a few comments about the Schur complement system.

Remark 3: By the Schur complement formulaiton the number of unknowns has been reduced, from what was essentially an N dimensional problem to an $(N-1)$ dimensional problem. This significant reduction in model complexity comes at the cost of the Schur complement system being in general much denser than the original system. Furthermore, the computational cost of calculating the full Schur complement matrices is frequently prohibitive. As such, the Schur complement formulation by itself is seldom used.

Remark 4: For local discretizations the direct coupling between variables p_i and p_j, where i and j are more than one integer apart, is usually small (and indeed there is no coupling for the two-point flux approximation methods). This implies that the matrices A_{ij} and thus also S_{ij} are for many practical problems essentially zero for $|i-j| \geq 2$, and the full Schur complement system is therefore essentially block tri-diagonal. Furthermore, we see that S_{ij} only differs significantly from A_{ij} in the case where $i = j = N-1$.

In the particular case of two spatial dimensions, the matrix S describes interaction between edge and vertex nodes only. In the case of three dimensions, it describes the interaction between faces, edges and vertexes, where we expect that the interactions between vertexes and faces are weak.

While the Schur complement system itself may be prohibitive to form and solve, it provides the framework for developing approximate solvers. Classically, these fall in the category of domain decomposition preconditioners (Smith, et al., 1996; Quateroni & Valli, 1999; Toselli & Widlund, 2005). In this chapter, we see how this framework also gives us both multiscale methods and preconditioners based on them.

Recall that the Schur complement system is (essentially) tridiagonal. The main approximation strategies to this system fall in two categories: The top down strategy gives a low-rank approximation to S based on only the degrees of freedom associated with vertexes of the coarse dual grid, which are then identified as the coarse degrees of freedom. This essentially forms a multiscale subspace based on the lower-diagonal component of S, and is the approach we will emphasize in the following. The bottom-up strategy goes the other way, successively applying Schur complement strategies to eliminate all variables until only a system for p_0 remains. Since the Schur complement matrices themselves are too expensive to calculate, the bottom-up approach requires introducing low-rank approximations to the Schur complement (e.g. probing based techniques (Chan & Mathew, 1992)) at every stage in the succession. The class of domain decomposition methods known as substructuring methods is often formulated in terms of the bottom-up framework.

3.2 Multiscale basis approximations

The multiscale basis approximations to the Schur complement system use the (block) lower diagonal component of S. Retaining the dependence on p_0, which we hereafter identify as the coarse variable, we then see that we obtain an explicit expression for the remaining degrees of freedom. In the block tri-diagonal case, this can be written compactly as:

$$p_i = \left(\prod_{j=1}^{i} -\hat{S}_{jj}^{-1} \hat{S}_{j(j-1)} \right) p_0 + \sum_{k=1}^{i} \left(\prod_{j=k+1}^{i} -\hat{S}_{jj}^{-1} \hat{S}_{j(j-1)} \right) \tilde{v}_k.$$

In this expression, the matrix products are ordered right to left, and we have marked the Schur complement matrices with a hat, indicating that approximate choices of these matrices can be used in order to define different multiscale bases. In the general case, where a block tri-diagonal system is not assumed, the above expression is defined recursively. Either way, for conciseness, we denote the linear operator associated with the reconstruction of the full approximation p by its homogeneous and heterogeneous parts,

$$p = \Psi p_0 + \Upsilon \tilde{v}.$$

At this point we make the following remarks.

Remark 5: The space spanned by the projection of p_0 to the full set of variables defined by Ψ is termed the *multiscale space* W_H^{MS}. It can be characterized by the basis functions obtained by setting $p_0 = e_i$, where e_i is the elementary vectors. The resulting product allows us to define the *multiscale basis function* ψ_i^{MS} as columns of Ψ,

$$\psi_i^{MS} \equiv \Psi e_i.$$

Given suitable choices of $\hat{S}_{ij}$, this gives various multiscale basis functions from literature, as seen in the following remarks. In the terminology of domain decomposition, these basis functions are often referred to as prolongation operators.

Remark 6: The natural interpretation of $\hat{S}_{jj}^{-1}\hat{S}_{j(j-1)}$ is to solve local problems at the level j using level $j-1$ as boundary conditions. This motivates the usual approximations to these Schur complements. Three important alternatives exist.

1. $\hat{S}_{jj}$ can be chosen as a discretization of the original differential operator restricted to the part of the internal boundary associated with j. This is the original multiscale basis functions of Hou & Wu 1997, and this is also the strategy we will apply in our numerical experiments.
2. For arbitrary operators, the differential operator restricted to a lower dimension may not be a good approximation to the problem, and this approximation is unstable. For such cases, a simple linear interpolation on internal boundaries can be suggested (Lunati & Jenny, 2007), and $\hat{S}_{jj}$ is then chosen as any matrix which admits the relevant (multi-)linear solutions.
3. Both the preceding operators require knowledge about the original geometry of the problem, and can thus be seen as geometric methods. If it is desired to implement multiscale methods strictly algebraically, then it is possible to construct algebraic approximations $\hat{S}_{jj}$ based on the information in S_{jj}, as was explored in Sandvin, et al., 2011.

Remark 7: It is common to not approximate the last Schur complement S_{NN}. Note that this does not imply that this Schur complement matrix needs to be computed, as we only need to know its action on the elements of the multiscale basis and on the right hand side. If this component is retained exactly, then the method becomes residual-free on the subdomains, which is an important aspect that can be exploited at later stages.

Keeping in mind that we now have an explicit representation of the solution covering the domain given the knowledge of the coarse nodes, we can use this representation to obtain a coarse system of equations.

3.3 Coarse scale equations retaining conservation form

From the last section, we see that we can use the Schur complement system to obtain a multi-scale basis. This is essentially a low-dimensional approximation of the solution space for the homogeneous part of the discrete differential operator. What remains in order to get an approximate solution is to consider coarse equations that constrain the remaining degrees of freedom in the multiscale space.

The original system of equations provides us with the first option for a set of coarse equations, since the equations associated with the coarse variables are simply our fine-scale discretization. Recalling the notation Ψ that indicates the linear operator that reconstructs the homogeneous part of the solution from the coarse basis we see that our original system of equations is simply

$$(A_{00} \quad \cdots \quad A_{0N})\Psi\, p_0 = v_0 - (A_{00} \quad \cdots \quad A_{0N})\Upsilon\tilde{v}.$$

This is however a poor choice of constraint for our coarse variables, as it physically represents only the differential operators locally around the coarse node. From the

perspective of the variational derivation of the control volume method, this solution thus satisfies Eq. (3) with $p \in W_H^{MS}$ and w in the space of piecewise constants with support around the *local cells associated with* p_0. From this understanding, we are motivated to think of reformulating the system such that the test functions for the coarse equations have a larger support, and in particular also form a partition of unity.

To be precise, consider a coarse partition of the domain, referred to as the primal coarse grid, which has the following properties: Each cell in the primal coarse grid consists of a set of cells from the fine grid, and contains exactly one vertex (cell on level 0) of the dual coarse grid, see Fig. 2. Since the primal coarse grid is a subset of the fine-scale grid, we know that the space of piecewise constant functions W_H on the primal coarse grid is a sub-space of the space of piecewise constant functions W_h on the fine grid. Therefore, by a change of representation, we could write the original discretization such that the discrete equations for the coarse variables satisfied Eq. (3) for all piecewise constant functions of the primal coarse grid. This leads to our desired coarse equations.

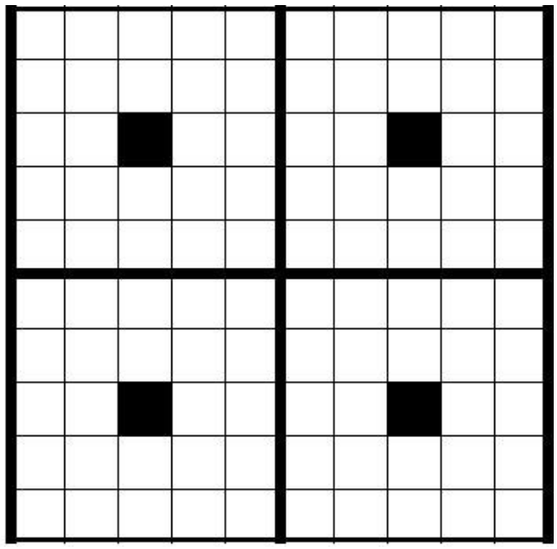


Fig. 2**.** A two-dimensional fine scale grid with a primal coarse grid imposed on it (bold lines). Black cells denotes center cells in the primal cell, these are on level 0 in the dual topology.

More practically, let the A, as before, represent a standard control volume discretization. Then let R_i be the restriction matrix to primal coarse cell i, and let $A_i = R_i A$. If furthermore M is an integration matrix that sums all rows in A_i into the row of the center cell, that is

$$M_i = I + e_{i0}(\mathbf{1} - e_{i0})^T, \tag{6}$$

where I is an identity matrix, e_{i0} is a unit vector identifying the center of the coarse cell, and $\mathbf{1}$ is a vector of ones. Multiplication with M_i for all primal coarse cells, and mapping the result back to the whole domain gives a linear system

$$Cp = v, \tag{7}$$

where $C = \sum_i R_i^T M_i R_i A$, and $v \longrightarrow \sum_i R_i^T M_i R_i v$. The linear system (7) is the original conservation of mass on the fine scale for all variables p_i where $i \geq 1$, however it represents

conservation on the coarse scale for variables p_0. Note in particular that this means that $C_{ij} = A_{ij}$ for $i \geq 1$, and that this linear transformation does not change the solution p.

We now see that the coarse equations, as given by

$$(C_{00} \quad \cdots \quad C_{0N})\Psi\, p_0 = v_0 - (C_{00} \quad \cdots \quad C_{0N})\Upsilon\tilde{v},$$

solve the problem given by Eq. (3) for with $p \in \operatorname{span} \psi^{MS}$ and $w \in W_H$, the space of piecewise constant functions on the dual coarse grid. We have thus derived a coarse control volume discretization, utilizing exactly a multiscale basis function to represent the solution. As a direct method, this is the so-called Multiscale Control Volume (Finite Element) Method as was first discussed (assuming $v_i = 0$ for $i \geq 1$) in Jenny, et al., 2003. The multiscale control volume methods described in the context of linear preconditioners are the Mass Conservative Domain Decomposition preconditioners derived in Nordbotten & Bjørstad, 2008.

Remark 8: In *Remark 1* at the end of the discretization section we saw that the standard finite element method is obtained by choosing test functions that are in the same multiscale space W_H^{MS} as the solution space. One may ask if the same is the case for multiscale methods. The answer is that yes, in the sense that if the integration on the primal grid defined in Eq. (7) is replaced by a weighted sum, using the multiscale basis itself as weights, the classical Multiscale Finite Element method of Hou & Wu, 1997 is recovered.

3.4 Recovering a conservative fine-scale flux field

The method as outlined so far constructs a two-level set of control volume methods. This can be seen from several perspectives: Either as the basis for a multi-level method, as the basis for a preconditioner in an iterative method, but also from the perspective of deriving a new, (coarse) single-scale control-volume method. We will consider the third perspective in this section.

When discussing the coupled set of equations outlined in Section 2.1, we pointed out the importance of retaining local mass conservation. This property is often necessary to consider (almost) point-wise, while by construction, the control-volume methods consider this only on the primal cells of the grid. It is therefore natural to consider whether a post-processing can be performed to extend this cell-wise property to a more local property, and whether this operation can be conducted locally.

For a local post-processing, it is natural to use the (cell-wise conservative) fluxes over boundaries of the primal grid as the basis of solving Neumann boundary problems inside each cell. The Neumann problem for the elliptic problems we consider is well-known to only admit solutions if the compatibility condition is satisfied, which is to say that the boundary conditions exactly integrate to the sum of all internal sources or sinks. The control-volume methods satisfy the compatibility condition by construction. Note that after post-processing, we will obtain a flux field that is everywhere conservative, but as a consequence will not everywhere satisfy Darcy's law.

In the case of single-scale control-volume methods, the permeability coefficient k is usually considered constant inside each primal cell, and locally post-processed fluxes can be

calculated analytically for some cell shapes. While this is not used much from the perspective of practical simulation, it is an invaluable tool in the derivation of error estimates.

For the multiscale control-volume method, the permeability is of course possibly heterogeneous inside each coarse primal cell, and a numerical calculation must be performed as a post-processing step. This can be achieved using the same grid and discretization as used when obtaining the multi-scale basis functions, and leads to an approximation with the following important properties: A post-processed flux which is conservative on the fine-scale primal grid. This post-processed flux allows for transport simulations to be performed on a significantly finer grid than the coarse control-volume scheme that was derived.

It is important to note that the possibility of post-processing the fluxes is the most important property of the multiscale control-volume method. Moreover, the construction of the MCDD preconditioner explicitly preserves this property, such that at any iteration of an iterative approach, the approximate solution to the fine-scale problem can also be post-processed in an identical manner.

3.5 Multiscale methods as iterative solvers

The domain decomposition method formulated in Section 3.3 can be applied as a stand-alone solver for the pressure system (4). This was the approach advocated in the early multiscale papers (Hou & Wu, 1997; Jenny, et al., 2003; Aarnes, 2004). Since the action of the method on a vector can be evaluated solving local systems related to the Schur complements, as well as a (relatively small) coarse linear system, we understand that the method offers an efficient way to obtain a pressure approximation and a mass conservative fine scale velocity field. Indeed, simulations of petroleum recovery indicate that in some cases, this strategy provides a fairly accurate and very cheap alternative to traditional approaches.

However, the above strategy is insufficient for more challenging problems. A particular weakness of the multiscale methods is the reliance on somewhat arbitrary approximations to the Schur complements $\hat{S}_{ii}$. Indeed, since the approximate Schur complements determine the subspace W_H^{MS}, we understand that for any approximate Schur complement, cases exist where the solution to the fine-scale problem lies in a space orthogonal to the multiscale space. Thus multiscale methods as direct solvers will always have problems with robustness. The practical performance of multiscale methods unfortunately deteriorates with the number of spatial dimensions; to be specific, multiscale methods have turned out to perform significantly worse in three spatial dimensions than in 2D.

When faced with these issues, there are a few techniques that can be applied to improve the solution. One is to consider sophisticated ways to construct $\hat{S}_{ii}$, using in particular non-local information and information about the right hand side. Another approach, to be described next, is to apply the MCDD preconditioners in an iterative setting to improve the approximation. The simplest such strategy is a Richardson scheme, where we, equipped with an initial guess p^0, define the iterative scheme

$$p^{l+1} = p^l + \tau B_{MS}(b - Ap^l),$$

where B_{MS} represent one application of the multiscale method and τ is a damping factor. We observe that when the multiscale method is applied as a stand-alone solver, this corresponds to applying a single Richardson iteration with the MCDD preconditioner and $\tau = 1$. The Richardson scheme will in general exhibit poor convergence for our problem. A better utilization of the multiscale method is as a preconditioner inside an iterative solver such as GMRES (Saad & Schultz, 1986). Since the problem is likely to be more difficult in some parts of the domain, the application of the preconditioner can be restrained to those parts, if they can be identified by error estimates.

An important and often time consuming ingredient of GMRES is to ensure orthogonality of the basis vectors for the Krylov subspace in which the approximated solution lies. When GMRES is preconditioned with the multiscale method, this computational cost can be reduced considerably by exploiting a special feature of the solution: If the internal nodes p_N are eliminated using an exact solver, e.g. introducing no approximation to S_{NN}, the residual in the interior will be zero after one application of the preconditioner as discussed in *Remark 7*. This does not mean the pressure is exact for those cells, but rather that the influence of nodes on level N on the residual is lumped into the higher levels. This also means that GMRES does not need to minimize the residual for cells on level N, the orthogonalization needs only consider levels $0, \ldots, N-1$, leading to an often significant reduction of the computational cost. Note that level N cannot be totally ignored, since some nodes there are a part of the flux expression for level $N-1$.

We now realize that the MCDD applied as a preconditioner in an iterative setting possess several advantageous features in comparison to standard preconditioners:

1. For relatively simple problems, where standard multiscale methods are applicable, the iterative procedure can be terminated after a single iteration.
2. For moderately complex problems, the iterative method can be terminated at any point where the solution is deemed accurate enough, and a locally conservative flux field can be recovered.
3. For truly challenging problems, the MCDD preconditioner is comparable to standard non-overlapping domain decomposition based preconditioners for these problems.

Thus we see, for applications where the exact solution to the linear system is not necessary, the current methodology allows for a substantial savings in number of iterations. This is of great practical importance, since the error introduced by a discrete approximation to (2) can frequently be orders of magnitude larger than the tolerance used in traditional linear solvers.

3.6 Computational cost

While a full assessment of the computational cost is beyond the scope of the chapter, we will make some brief comments that allow the reader to get a general impression of the cost of both the multiscale methods as well as their application as preconditioners.

The computational cost of the MCDD preconditioner is composed of three components. First, the approximate Schur complement system involves approximating the action of S_{NN} on a (small) set of vectors. Physically, this corresponds to solving the local problems inside the internal subdomains for given boundary conditions. Denoting the number of internal

subdomains as N_{SD}, a naïve estimate of cost would be $N_{SD} \cdot \dim p_0$. However, by construction, most approximate Schur complements will be local, such that each subdomain typically only has non-zero boundary conditions associated with the variables in p_0 that are associated with cells on the boundary of the subdomain. For Cartesian coarse subdomain, this is identified as the corners, such that the computational cost is proportional to $N_{SD} \cdot 2^d \cdot C_F$. Here C_F is defined as the coarsening factor, which is the ratio of degrees of freedom in the fine and coarse spaces, $C_F \equiv \frac{\dim p}{\dim p_0}$. The multiscale basis is only calculated once.

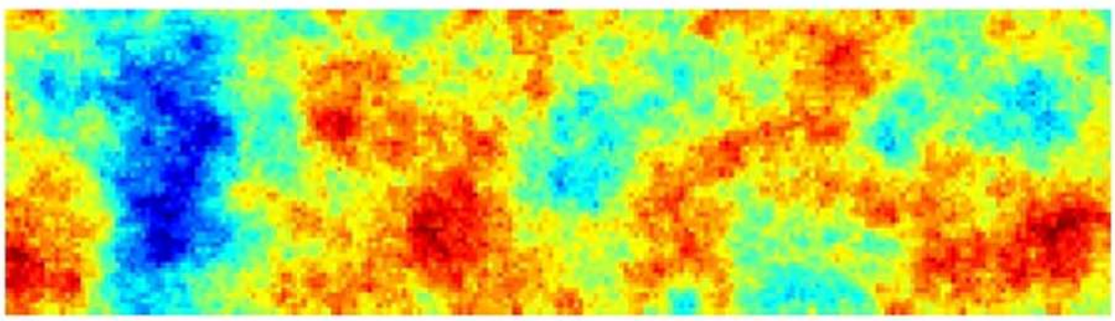
Secondly, there is a cost associated with the right-hand side, which needs to evaluated at every iteration. As seen in Section 3.2, the right hand side is also associated with local calculations, forced by source terms in contrast to the multiscale basis functions. The cost is thus proportional to $N_{SD} \cdot C_F \cdot N_I$, where N_I is the number of iterations.

Finally, there is the cost associated with solving the coarse set of equations. Here, there are two contrasting strategies. The domain-decomposition strategies argue for aggressive coarsening, where the coarse problem has (almost) negligible size and cost. This has the advantage that the cost of the coarse solve can be neglected, at the expense of more costly construction of the multiscale basis. However, as the multiscale basis calculation is trivially parallel, this may be a good strategy on some computational architectures, and in particular if the selection of coarse grids is hard to automate. A contrasting strategy is in the multi-grid flavor, where a much less aggressive coarsening is applied, which leads to a non-negligible cost in the coarse problem. However, since the coarse problem has the same control-volume structure as the fine-scale discretization, the multiscale method can be called recursively. The resulting algorithm has a better performance from the perspective of computational cost, but may be more difficult to implement as the problem is no longer trivially parallel. Note that for a conservative approximation to be obtained, the reconstruction of the flux field must also be conducted recursively.

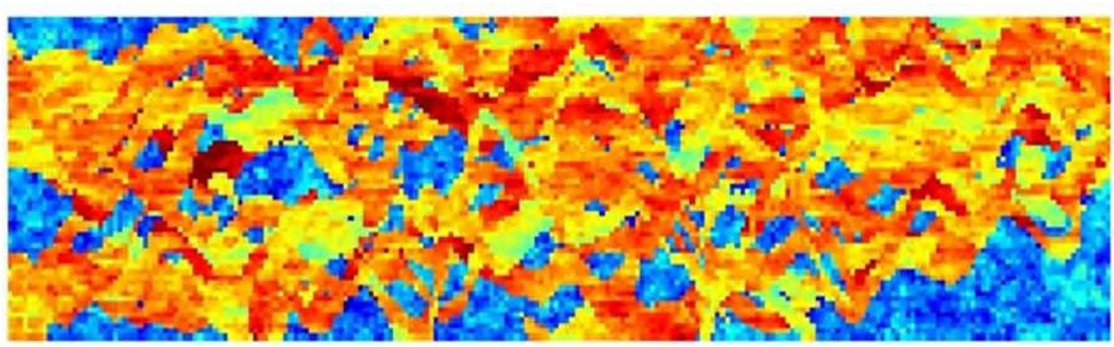
In general, the multiscale methods are designed for problems where there is a coupling between the permeability k and the concentration field c. As c evolves locally, the multiscale basis functions may only need to be updated locally in space, allowing for further computational savings compared to a generic linear solver that is not adapted to these features. These aspects have been carefully highlighted in a suite of 2D test cases (Kippe, et al., 2008).

4. Numerical examples

In this section, we show numerical examples illustrating the properties of the domain decomposition method. For these examples, we have chosen the permeability field defined according to the SPE 10th comparative benchmark study much used to study upscaling and multiscale methods (Christie & Blunt, 2001). This test case involves a Cartesian 60 x 220 x 85 grid. The permeability in the upper 35 layers have a somewhat smooth distribution (consistent with a shallow marine depositional system), whereas the 50 lower layers are characterized by sharp permeability contrasts and highly permeable channels with long correlation length (consistent with a fluvial depositional system). The lower layers are expected to pose challenges for linear solvers. Representative layers from the upper and lower parts of the formation are shown in Fig. 3. The permeability field spans more than 10 orders of magnitude, rendering a challenging test problem for our methods.



(a) Uppermost layer



(b) Lowermost layer

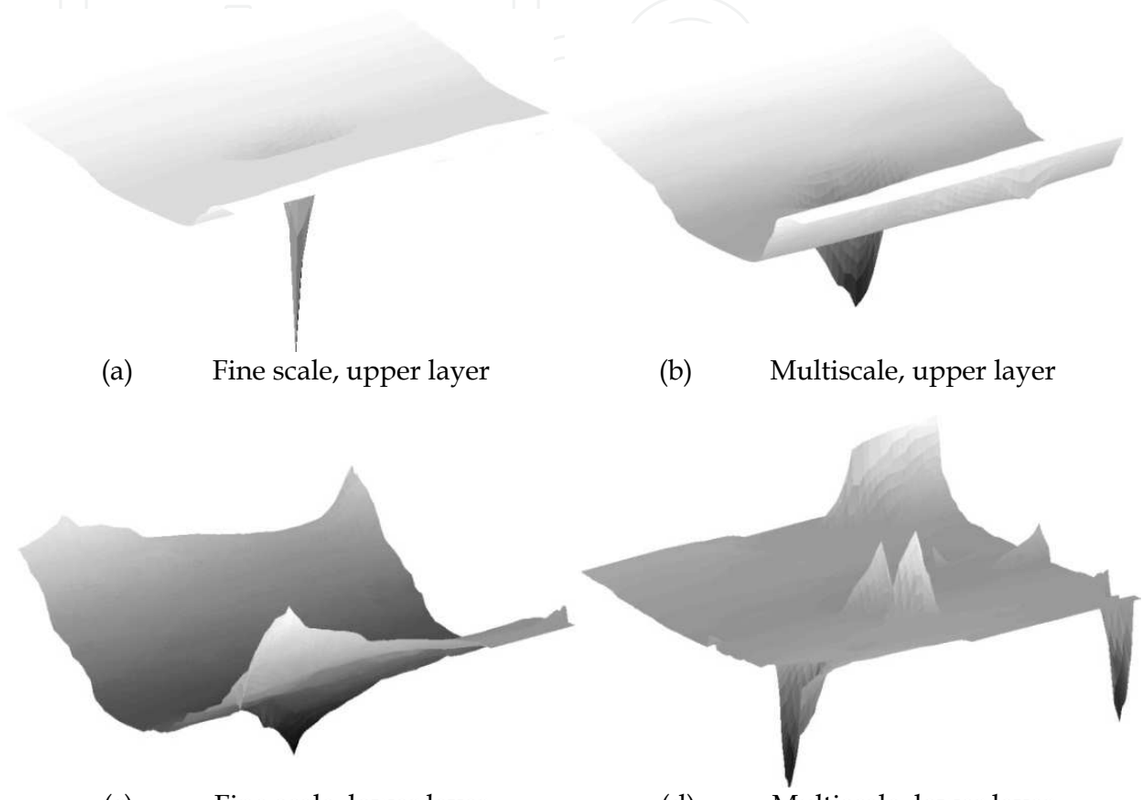
Fig. 3. The base-10 logarithm of the permeability from the uppermost (a) and lowermost (b) layers in the SPE 10 test case. These are used in the 2D tests, and they are representative for the upper and lower part of the 3D formation, respectively. Blue and red corresponds to high and low-permeable regions, respectively. For convenience, the figures are rotated $\mathbf{90^o}$.

On this grid, we will consider simple setups, with one injection well and one producer. Both for 2D and 3D tests the injector is located along the boundary (the position differs somewhat between the tests, as we avoid injecting into low-permeable cells), and the producer is located in the middle of the domain. The pressure equation (2) is discretized using a two-point scheme, and for simplicity, periodic boundary conditions are assumed. For all test cases, post-processing of the flux field as discussed in Section 3.4 will be applied to ensure mass conservation on the fine scale.

4.1 2D examples

We start with two instructive examples in 2D, using permeability from the uppermost and lowermost layer of the SPE10 dataset, as pictured in Fig. 3. Thus the fine scale grid has 60 x 220 cells, and we use a coarse grid with 4x20 cells, rendering a coarsening factor of 165. Fig. 4 shows the pressure profiles obtained by a fine scale solution and the multiscale solver. For the upper layer, the multiscale solution is similar to that of the true solution; and has a quality that is as good as can be expected keeping in mind that the multiscale method is essentially a coarse discretization. In both solutions the pressure contours clearly

indicate flow from injector to the producer, although again, the resolution of the local flow around the producer is better refined on the fine-scale grid. For the lower layer, the multiscale solution is highly oscillatory with false local minima in the solution. This can be interpreted as a case where the approximation to the Schur complement $\hat{S}_{ii}$ is not good enough, where a better approximation, or iterations, are needed to produce a pressure profile that resembles that of the fine scale solution.



(a) Fine scale, upper layer (b) Multiscale, upper layer

(c) Fine scale, lower layer (d) Multiscale, lower layer

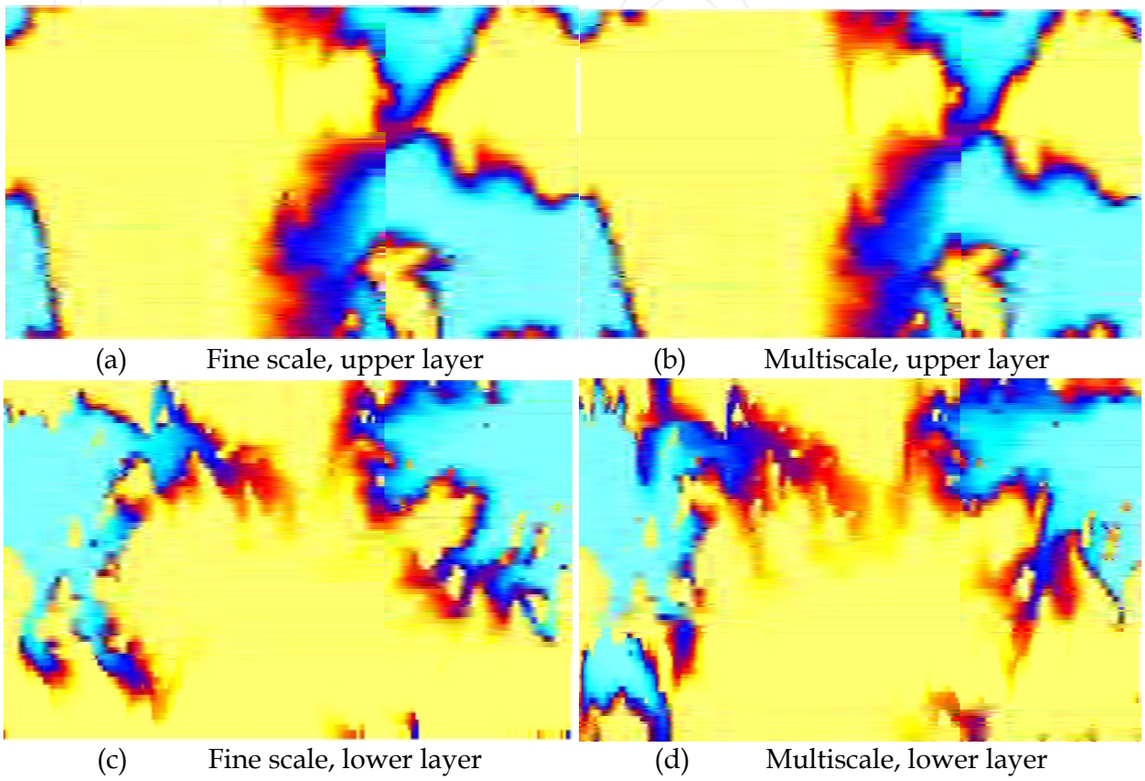
Fig. 4. Pressure solutions obtained by a fine scale and a multiscale solution for the uppermost and lowermost permeability layers. The injection well is located along the left boundary for all plots, while the producer is associated with the downward spike visible in the middle of the domain visible in all figures except (d).

For the uppermost layer of SPE10, the relatively good MS approximation to pressure is reflected in the post-processed fluxes. We illustrate this by the solution to the transport equation (1), as displayed in Fig. 5 (a) and (b). Note that despite the relatively coarse grid used for the multiscale control-volume approximation, the reconstruction of the fine-scale fluxes leads to a flow field with no visible artifacts. From the perspective of practical simulation, the solutions are indistinguishable.

Surprisingly, despite the relatively poor approximation to the pressure field, quite satisfactory fluxes can be obtained also for the lowermost layer as shown in Fig. 5 (c-d). This illustrates that the coarse scale conservation of mass combined with post-processing of the velocity field, leads to a multiscale approximation that is applicable to transport problems also for highly challenging problems. Note however that in these lower layers,

the multiscale approximation leads to some cases where flow-channels are either suppressed or exaggerated.

The results from the concentration maps in Fig 5 are further confirmed by considering time series of the concentration in the production well, as shown in Fig. 6. For the upper layer, the curves corresponding to the fine scale and multiscale solutions are almost identical, and the differences are relatively small also for the lower layer.

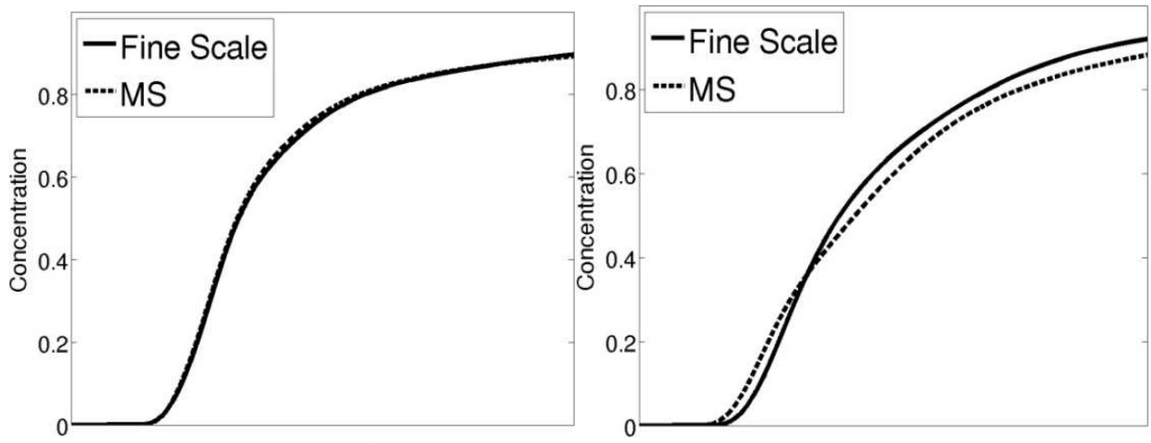


(a) Fine scale, upper layer (b) Multiscale, upper layer

(c) Fine scale, lower layer (d) Multiscale, lower layer

Fig. 5. Concentration profiles obtained by solving the transport equation based on the post-processed pressure solutions. High concentration of the injected species is indicated with blue. We emphasis that for the multiscale solution, the velocity field is post processed to achieve local conservation of mass.

Remark 9: The appearance of oscillatory behavior in the multiscale solution is not unexptected. Again, we can analyze the multiscale control-volume method as simply being a single-scale control volume method on the coarse primal grid. It is known that for problems where the anisotropy in k is not aligned with the grid, local control-volume methods (and indeed this also holds for some other discretization families) in general cannot be constructed that are both consistent, as well as oscillation-free (Nordbotten, et al., 2007; Keilegavlen, et al., 2009). The channelized features that are shown in Fig. 3b are clearly not aligned with the general directions of the domain, and therefore they will lead to an effective permeability on the coarse grid that is also not aligned with the grid. The argument from the single-scale methods can thus be lifted to the multi-scale setting, which then informally may be states as: *No approximation $\hat{S}$ can be defined that leads to a local coarse-level control-volume method that is monotone for general channelized media.*



(a) Upper layer (b) Lower layer

Fig. 6. Time series of the concentration in the production well in the upper and lower layer.

4.2 3D examples

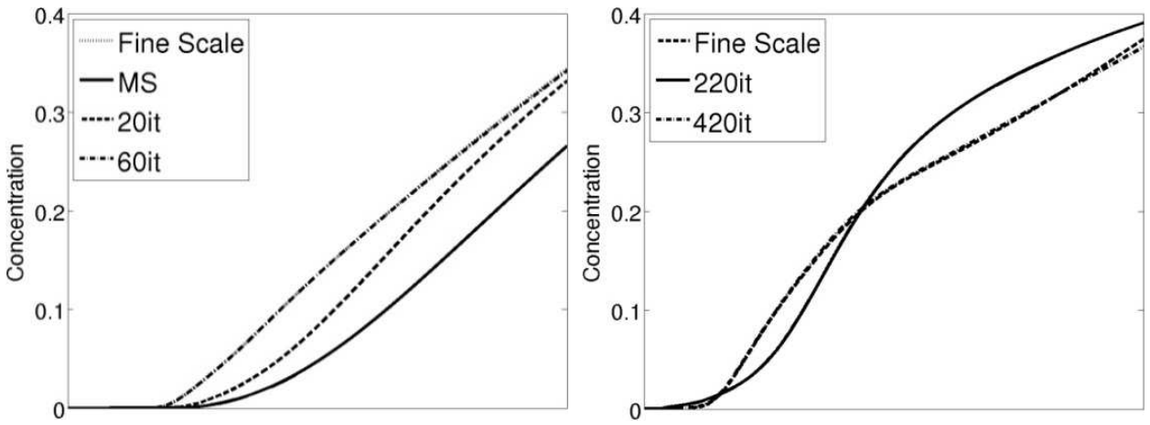
The 2D examples showed that the multiscale method can provide reliable solutions for challenging permeability fields and relatively high coarsening ratios. As previously mentioned, the performance of the multiscale method deteriorates significantly when going from 2D to 3D. As we will see, the multiscale solution may be insufficient for transport purposes, and the application as a MCDD preconditioner inside an iterative solver is essential in order to recover accuracy. For all 3D simulations, we consider coarse grid cells composed of 15 x 11 x 5 fine cells, rendering a coarsening ratio of 825.

4.2.1 Multiscale method as preconditioner

We first consider simulations in the 10 uppermost and lowermost layers of the SPE10 formation, extending the two cases considered in the 2D case. Again there is an injection well in a corner of the domain, and a producer in the middle of the domain. We consider transport solutions based on a fine scale solution, a pure multiscale solution, and from MCDD preconditioned GMRES iterations. Since visualization is more difficult in 3D than in 2D, we will in 3D only give the time-series type plots similar to Figs. 6. The time series of concentration in the production cells are shown in Fig. 7 both for the upper and lower layers. For the upper layers, we observe that in contrast with the 2D examples, the multiscale solution now deviates significantly from the fine scale solution. Applying some GMRES iterations improves the quality of the solution somewhat, until after a sufficient number of iterations renders a curve that is indistinguishable from the fine scale. For the lower layers, the stand-alone multiscale solver produce a time series that is vastly different from the fine scale solution, and it is therefore not shown in the figure. For this difficult problem, it takes more iterations to produce a time series that resembles that of the fine scale solve.

The above test shows that in 3D the multiscale method does not reproduce concentration curves that are comparable to the fine-scale curves even when for relatively easy case of the upper layers of SPE10. Thus the present test shows the utility of having a framework that, when the fast multiscale solution is insufficient, can fall back to an iterative scheme, and fairly quickly recover a velocity field that is good enough for transport purposes. The

increased difficulty in approximating the solution is also shown in the development of the residual during the corresponding GMRES iterations, see Fig. 8. In the lower layers the residual decreases slower and more iterations are needed to obtained what might be deemed a satisfactory solution. Based on these two simple tests, we observe that the relative residual error in the linear solver of 10^{-3} to 10^{-4} is needed in order to reproduce a good transport solution. This value is many orders of magnitude higher than the typical residual errors used in iterative solvers for the linear system. Indeed, the residual error as a function of iterations is shown in Fig. 8, where we see that more than 600 iterations are needed to obtain a converged iteration for the upper layers. The ability to truncate the iterations early when using MCDD as a preconditioner will therefore in this case represent a savings of about 90% in terms of number of iterations in the iterative solver. Also for the lower layers, an early truncation saves a majority of the computational effort.



(a) Upper layers (b) Lower layers

Fig. 7. Time series of the concentration in the production well for simulations in the upper and lower part of the SPE10 formation. The solutions obtained from the fine scale are located on top of those from 60 and 420 GMRES iterations for the upper and lower parts, respectively. A multiscale solution is not shown for the lower part, due to the low quality of the results produced.

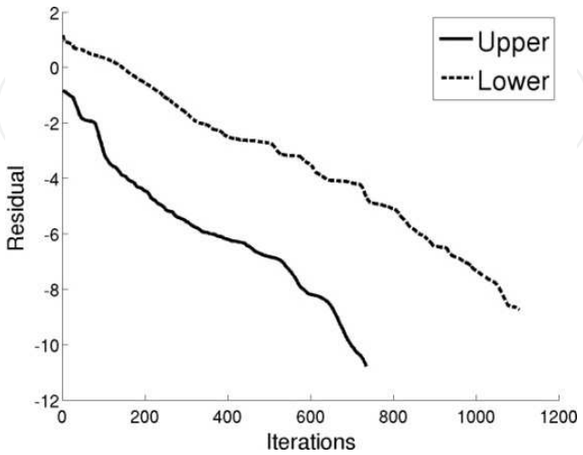


Fig. 8. The residual as a function of GMRES iterations for parts of the upper and lower part of the formation. The iterations terminates when the relative residual is reduced to a factor $\mathbf{10^{-10}}$.

At this point, it is appropriate to mention a third option to improve the multiscale solution, in addition to advanced approximations of the Schur complements $\hat{S}_{ii}$ and increasing the number of iterations: By increasing the number of coarse variables (in essence moving variables to level 0), the range of the multiscale basis functions ψ^{MS} can be increased to capture more of the solution. These ideas are exploited in (Sandvin, et al., Submitted), and show promising results in that the number of iterations needed can be reduced significantly with only a minor increase in the computational cost.

4.2.2 Quality control of MCDD solution

Returning to the original formulation of the system, we realize that an MCDD-based solution can be interpreted as the *exact* solution of Equations (2), for a modified permeability k^*. Using the pressure solution from the iterative solver together with the post-processed fluxes, we can calculate k^*. As the permeability is typically a value associated with great uncertainty for geological applications, we can compare the difference between k and k^* as a metric on the quality of the MCDD-based approximation. The most basic version of this comparison is to recall that from the physical motivation of the problem, the original permeability k is symmetric positive definite, and we can thus assess the quality of the approximate solution based on whether the modified permeability k^* also satisfies this physical constraint.

In Table 1, iteration counts and the number of sign changes are shown for a series of residual tolerances for GMRES. The grid consists of 60x220x10 cells, and the permeability is found from the channelized part of the SPE10 formation. Note that for the SPE10 dataset, the permeability tensor is diagonal, so positive definiteness is equivalent to positive diagonal elements. The table shows that for a high residual tolerance, more than a third of the fluxes change sign during post processing. Moreover, even for high accuracy of the GMRES solution there are some sign changes during flux post-processing.

$\log_{10}$(Tolerance)	-2	-4	-6	-8	-10
Iterations	224	429	717	915	1079
Negative elements in k^*	37%	9.4%	1.8%	.17%	.0018%

Table 1. The relative residual in GMRES, together with numbers for iterations and percentage of negative elements in the modified permeability.

The deviation of the flux and potential from a physical flow field, as measured by k^*, represents one attractive metric for assessing the approximation quality. However, more classical *a posteriori* error bounds and estimates are also applicable in this setting, and may be of equal importance for practical applications.

5. Concluding remarks

The present chapter has reviewed the construction of multiscale control volume methods in arbitrary dimensions from an algebraic perspective, allowing for a completely decoupled

implementation of the fine-level (control volume) discretization and the multi-scale framework. We have emphasized several important aspects, including the points where key approximations are made, together with both their algebraic and physical interpretations. By bringing attention to the formulation of multiscale methods in this general setting, we have been able to highlight aspects of how multiscale control volume methods relate to classical single-scale discretizations, iterative preconditioners, and multi-level approximations. Through carefully chosen numerical examples, we have sought to illustrate both the quality of the multiscale approximation to the primary variable (pressure), but more importantly the role of the multiscale approximation in the setting of a coupled system of equations. These examples clearly illustrate the increasing complexity faced with problems in 3D over 2D, and the care with which one needs to deal with notions of approximate solvers and multiscale numerics.

In closing this chapter we wish to take the opportunity to discuss some of the main obstacles and benefits of multiscale methods as one considers more challenging problems.

As a stand-alone solver for a single elliptic problem, it is difficult for multiscale methods and preconditioners to compete with multigrid methods. The advantage of the methodology lies therefore in different aspects.

- A coarse discretization is obtained directly, with explicit coarse flux expressions, leading to an understanding of the nature of the effective coarse-scale operator for the system.
- For time-dependent, where multiple (similar) problems need to be solved in succession, a large amount of calculations can be re-used from previous time-steps.
- For (locally) spatially periodic problems, sub-domain problems may be identical and computational savings can be obtained through re-use again.
- For problems with scale-separation (where homogenization is applicable), the multiscale method gives a good approximation to both the homogenized and true solutions after a single iteration.

Despite the initial promise, and the evidence that the advantages can be realized for model problems, several challenges remain before multiscale methods attain the robustness required for practical applications. Some of the major limitations, together with their potential remedies are:

- For irregular grids (both on the fine and coarse scale) and for anisotropic media, the multiscale approximation is again less robust, especially when local Schur approximations are applied. To some extent, this can be overcome by oversampling, through enriching the coarse space, or by bottom-up approaches such as matrix probing, although as noted in *Remark 9*, a local and consistent coarse operator can in general never be designed.
- For non-linear elliptic equations (e.g. if the permeability coefficient is a function of the pressure or its gradients), the method is no longer residual-free in the interior if the multiscale basis functions are re-used. Recalculating multiscale basis functions in an iterative setting is prohibitively expensive, and it remains unclear if good multiscale approximations can be constructed.
- For higher-dimensional problems (more than 3), the quality of multiscale approximations has yet to be addressed at all.

With these perspectives in mind, it is clear that multiscale methods and preconditioners are still a topic of very active research. As such, there will most certainly be aspects of the current chapter that later research will both clarify and improve upon. Nevertheless, we hope that the present text succeeds in giving a current perspective on multiscale methods that will have value for both the general and specialized reader.

6. Acknowledgement

This research was financed in part through the Norwegian Reseach Grant #180679, "Modelling Transport in Porous Media over Multiple Scales". The book chapter is written in part based on lectures at the Radon Institute for Computational and Applied Mathematics (RICAM) Special Semester on Multiscale Simulation & Analysis in Energy and the Environment, Linz, October 3-December 16, 2011

7. References

Aarnes, J., 2004. On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation. *SIAM Multiscale Model. Simul.,* 2(3), pp. 421-439.

Aavatsmark, I., 2002. An introduction to the multipoint flux approximations for quadrilateral grids. *Comput. Geosci.,* 6(3-4), pp. 405-432.

Aziz, K. & Settari, A., 1979. *Petroleum Reservoir Simulation.* s.l.:Chapman & Hill.

Bear, J., 1972. *Dynamics of Fluids in Porous Media.* s.l.:Elsevier.

Chan, T. & Mathew, T., 1992. The interface probing technique in domain decomposition. *SIAM J. Matrix Anal. Appl.,* 13(1), pp. 212-238.

Chen, Z., Huan, G. & Ma, Y., 2006. *Computational methods for multiphase flows in porous media.* s.l.:SIAM.

Christie, M. & Blunt, M., 2001. Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *Proc. of SPE Reservoir Simulation Symposium.*

Edwards, M. & Rodgers, C., 1998. Finite volume discretization with imposed flux continuity for the general tensor pressure equation. *Comput. Geosci.,* 2(4), pp. 459-490.

Efendiev, Y. & Hou, T., 2009. *Multiscale finite element methods: theory and applications.* s.l.:Springer.

Hou, T. & Wu, X.-H., 1997. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.,* 134(1), pp. 169-189.

Jenny, P., Lee, S. & Tchelepi, H., 2003. Multi-scale finite-volume method for elliptic problems in subsurface flow simulations. *J. Comput. Phys.,* 187(1), pp. 47-67.

Keilegavlen, E., Nordbotten, J. & Aavatsmark, I., 2009. Sufficient criteria are necessary for monotone control volume methods. *Appl. Math. Lett.,* 22(8), pp. 1178-1180.

Kippe, V., Aarnes, J. & Lie, K.-A., 2008. A comparison of multiscale methods for elliptic problems in porous media flow. *Comput. Geosci.,* 12(3), pp. 277-298.

Lunati, I. & Jenny, P., 2007. Treating highly anisotropic subsurface flow with the multiscale finite-volume method. *SIAM Multiscale Model. Simul.,* 6(1), pp. 308-318.

Lunati, I., Tyagi, M. & Lee, S., 2011. An iterative multiscale finite volume algorithm converging to the exact solution. *J. Comput. Phys.,* Volume 230, pp. 1849-1864.

Nordbotten, J., Aavatsmark, I. & Eigestad, G., 2007. Monotonicity of control volume methods. *Numer. Math.,* 106(2), pp. 255-288.

Nordbotten, J. & Bjørstad, P., 2008. On the relationship between the multiscale finite-volume method and domain decomposition preconditioners. *Comput. Geosci.,* 12(3), pp. 367-376.

Quateroni, A. & Valli, A., 1999. *Domain decomposition methods for partial differential equations.* s.l.:Oxford University Press.

Saad, Y. & Schultz, H., 1986. A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.,* 7(3), pp. 856-869.

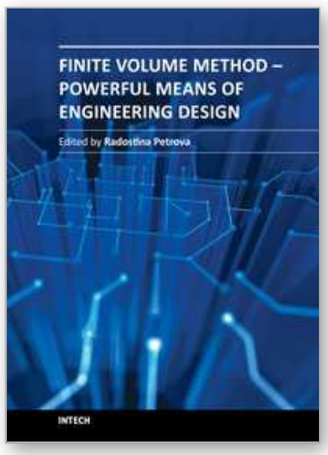
Sandvin, A., Keilegavlen, E. & Nordbotten, J., Submitted. Auxiliary variables in multiscale simulations. *J. Comput. Phys..*

Sandvin, A., Nordbotten, J. & Aavatsmark, I., 2011. Multiscale mass conservative domain decomposition preconditioners for elliptic problems on irregular grids. *Comput. Geosci.,* 15(3), pp. 587-602.

Smith, B., Bjørstad, P. & Gropp, W., 1996. *Domain Decomposition.* s.l.:Cambridge University Press.

Tchelepi, H. & Juanes, R., 2008. *Special issue on multiscale methods for flow and transport in heterogeneous porous media.* 3 ed. Comput. Geosci.: s.n.

Toselli, A. & Widlund, O., 2005. *Domain Decomposition Methods - Algorithms and Theory.* s.l.:Springer.



Finite Volume Method - Powerful Means of Engineering Design
Edited by PhD. Radostina Petrova

ISBN 978-953-51-0445-2
Hard cover, 370 pages
Publisher InTech
Published online 28, March, 2012
Published in print edition March, 2012

We hope that among these chapters you will find a topic which will raise your interest and engage you to further investigate a problem and build on the presented work. This book could serve either as a textbook or as a practical guide. It includes a wide variety of concepts in FVM, result of the efforts of scientists from all over the world. However, just to help you, all book chapters are systemized in three general groups: New techniques and algorithms in FVM; Solution of particular problems through FVM and Application of FVM in medicine and engineering. This book is for everyone who wants to grow, to improve and to investigate.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Jan M. Nordbotten, Eirik Keilegavlen and Andreas Sandvin (2012). Mass Conservative Domain Decomposition for Porous Media Flow, Finite Volume Method - Powerful Means of Engineering Design, PhD. Radostina Petrova (Ed.), ISBN: 978-953-51-0445-2, InTech, Available from: http://www.intechopen.com/books/finite-volume-method-powerful-means-of-engineering-design/mass-conservative-domain-decomposition-for-porous-media-flow

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen